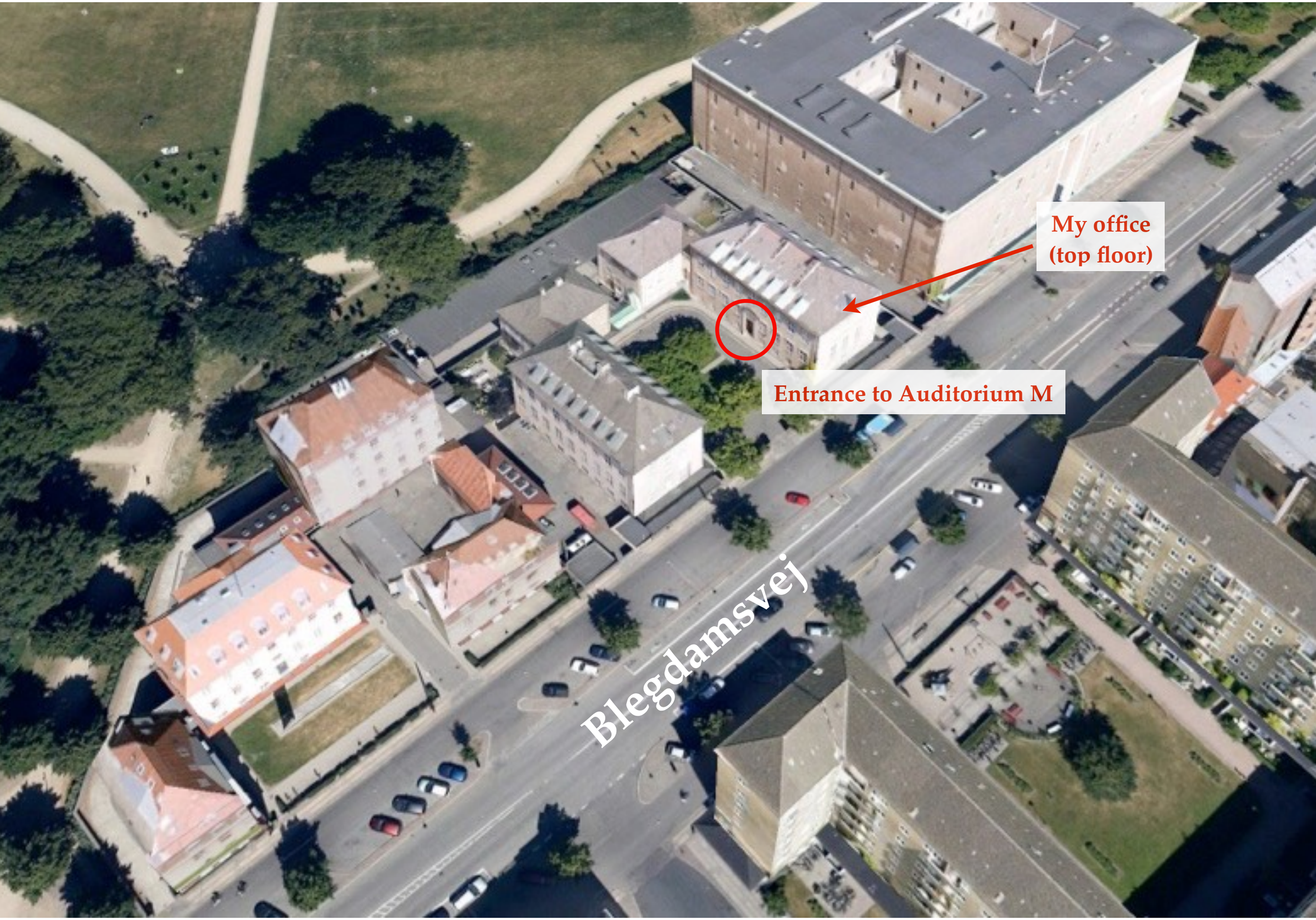
Course Information

D. Jason Koskinen
koskinen@nbi.ku.dk

Advanced Methods in Applied Statistics
Feb - Apr 2018

Photo by Howard Jackman


My office
(top floor)
Entrance to Auditorium M
Blegdamsvej

Times & Locations

From 08:30-09:00 I will be in Aud. M
but I will only be there for discussion and
student Q&A, i.e. no new material until 09:00

- Hours
  - Course is in Block A
  - Tuesday 08:00 - 12:00
  - Thursday 08:00 - 12:00 and 13:00 - 17:00
- Location: Auditorium M at Blegdamsvej

Actual
08:00-08:30 Study
08:30-09:00 Study/Q&A
09:00+ Lecture

- In-class Activities
  - ~20-30% of the time will be lecture
  - **Vast majority** of the time will be practical exercises
    - Finding appropriate software package or function
    - Properly instantiating the relevant statistical method
    - Debugging
    - Documentation, plotting, code clean-up, maybe even in-line comments, etc.

Me

- I go by “Jason”
- My scientific focus is on experimental neutrino oscillation, where I work on the IceCube neutrino observatory situated at the South Pole

Computers & Software

- Everyone should have a laptop
- Software specifics are left to the student
  - Suggestions are python, R, MATLAB, or C/C++
    - The more common the language the more likely you can use the internet and fellow students (possibly) for help
    - Lectures and examples will be mostly in python using some external packages:
      - ROOT (https://root.cern.ch), note that ROOT is *not necessary* and can be painful to install
      - SciPy (http://www.scipy.org)
      - NumPy (http://www.numpy.org)
- I encourage you to find at least one person who is nominally using the same setup, e.g. Windows 10, R 3.2.3

Software, Checklist, & Skills

- Have an installed text editor for writing/editing software
- Have some package for the production of plots and diagrams (ROOT, matplotlib, R, gnuplot, MATLAB, etc.)
  - See backup slides for some more specific software packages
- I strongly recommend reviewing the undergraduate stats course (http://www.nbi.dk/~petersen/Teaching/AppliedStatistics2017.html)
  - Actually do the exercises, not just scan the material, and see what isn't clear or familiar

Course Material

- **NO** required text or textbooks. I will cover many topics w/ in-class lectures and all the notes will be posted online. But, this may be insufficient in depth or explanation for your personal preference, so students are encouraged to use…
  - The Internet - probably the best source for information and help.
  - "Statistical Data Analysis" by Glen Cowan
  - "Modern Statistical Methods for Astronomy" by Feigelson & Babu
  - Journal articles
    - Any that you might find relevant
    - Some posted by me

Student Assessment

- Oral Presentation and 2-page summary (10%)
  - Take topic and/or relevant article for presentation to the class
  - Can be done in groups, but no more than 3 people

Student Assessment

- Oral Presentation and 2-page summary (10%)
  - Take topic and/or relevant article for presentation to the class
  - Can be done in groups, but no more than 3 people
- Graded problem sets (20%)
  - Can be done in groups of any size, but must be submitted individually

Student Assessment

- Oral Presentation and 2-page summary (10%)
  - Take topic and/or relevant article for presentation to the class
  - Can be done in groups, but no more than 3 people
- Graded problem sets (20%)
  - Can be done in groups of any size, but must be submitted individually
- Project (30%)
  - Larger data analysis project, nominally related to your field of research
  - Can be done in groups (no more than 3 people) with a single 4-6 page written report

Student Assessment

- Oral Presentation and 2-page summary (10%)
  - Take topic and/or relevant article for presentation to the class
  - Can be done in groups, but no more than 3 people
- Graded problem sets (20%)
  - Can be done in groups of any size, but must be submitted individually
- Project (30%)
  - Larger data analysis project, nominally related to your field of research
  - Can be done in groups (no more than 3 people) with a single 4-6 page written report
- Final Exam (40%)

Problem Sets

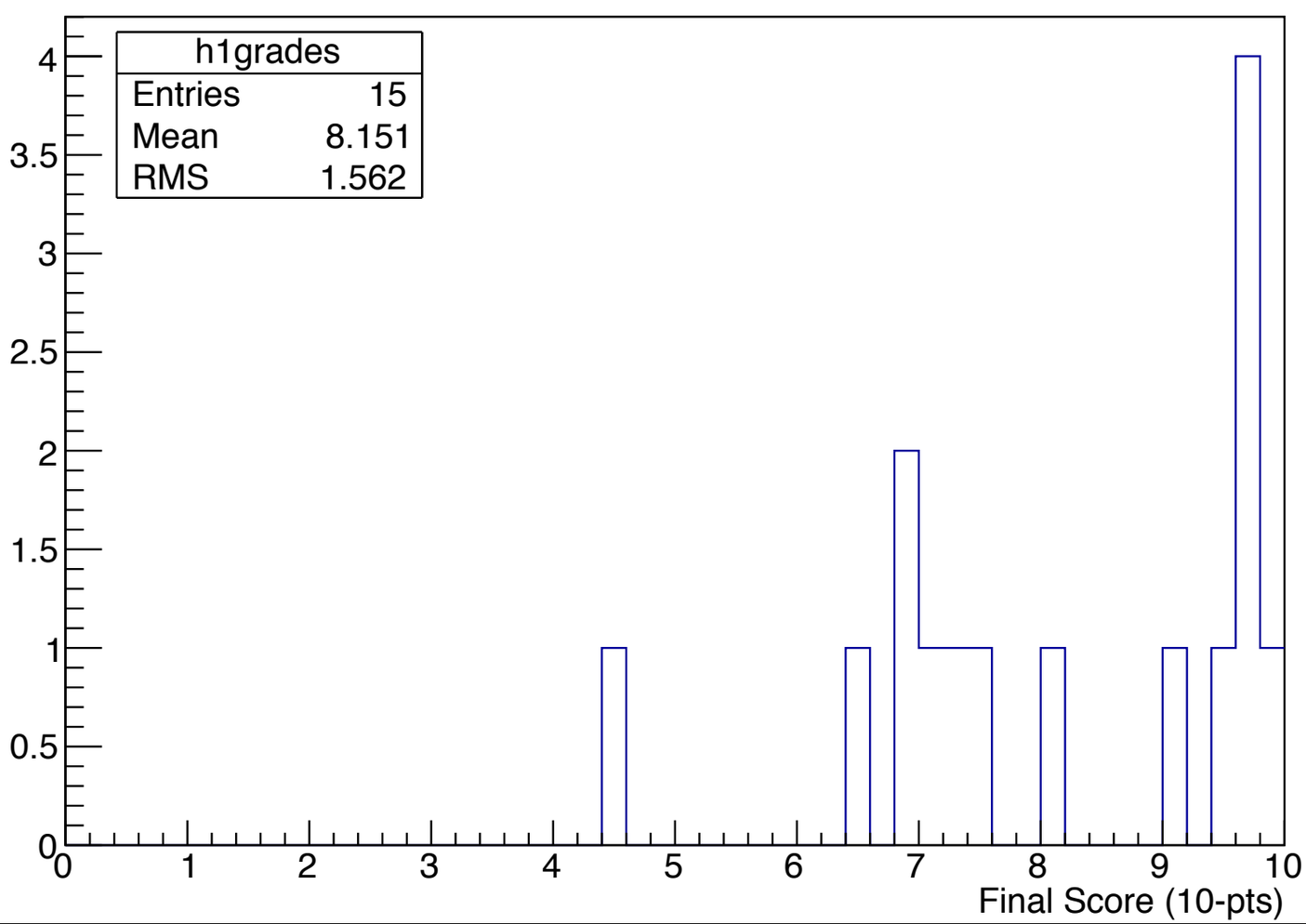
- There will be 2 or 3 problem sets to be handed in
- The submission is:
  - A write-up as a PDF document, which includes any plots, diagrams, tables, pictures, and explanations
  - In a separate “file”, submit all code used to derive the results
    - Tarball, zipped directory, lots of individual files w/ self-explanatory titles, etc.
    - Include original data files if possible
- Material is marked on a 10-point scale
  - 9+ is very good
  - 8-9 is pretty good
  - 7-8 is okay
  - 6-7 is acceptable
  - 5-6 subpar
  - 4-5 inadequate
  - <4 reflects serious omissions and/or deficiencies

Final Exam

- 1-day (~28 hour) take home test
- Requires computers, writing/modifying code that has been developed during the course
- **You must work on your own!!!**
  - Along with the answers, the code producing the results must also be submitted
- The exam will be available on April 5, 2017 with a submission deadline on April 6, 2017
  - Doing the in-class exercises and homework is excellent preparation for the exam
  - Start at 10:00 CET on Thursday and submit by 14:00 CET the next day.

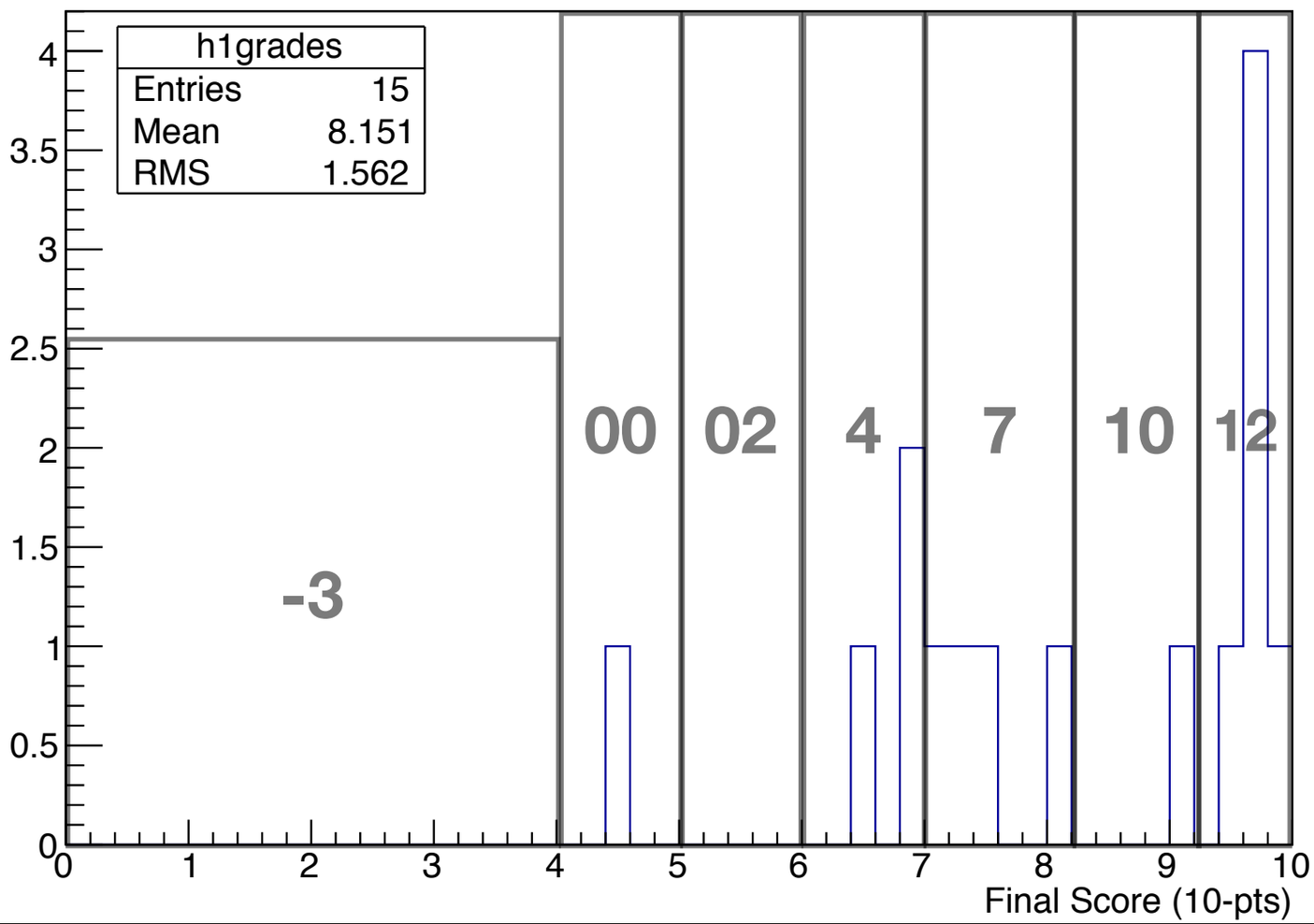
Grading Breakdown

- The final course grade is **converted** to the Danish 7-point scale, but all course material uses a 10-point scale. Last year it was an approx. decile conversion.
- The conversion will be ‘roughly’ decile, e.g. a 9.32 final weighted average is very, very likely to get a mark of “12”. But, 9.2 was the cutoff between a “10” and “12” last year, and 8.2 for between “7” and “10”.



Grading Breakdown

- The final course grade is **converted** to the Danish 7-point scale, but all course material uses a 10-point scale. Last year it was an approx. decile conversion.
- The conversion will be ‘roughly’ decile, e.g. a 9.32 final weighted average is very, very likely to get a mark of “12”. But, 9.2 was the cutoff between a “10” and “12” last year, and 8.2 for between “7” and “10”.



Student Assessment

- All assessment material will be graded based on the results
  - Code can be sloppy and inefficient and it is unlikely to impact your grade. The exception is where I'd like to give the 'benefit of the doubt', but can't decipher your code.
  - Obviously, the exception is cheating, e.g. using the 97% of the code of someone else and only changing the comments , variable names, line colors, etc.

Student Assessment

- All assessment material will be graded based on the results
  - Code can be sloppy and inefficient and it is unlikely to impact your grade. The exception is where I'd like to give the 'benefit of the doubt', but can't decipher your code.
  - Obviously, the exception is cheating, e.g. using the 97% of the code of someone else and only changing the comments , variable names, line colors, etc.
- I encourage you to share solutions, efficient code, elegant solutions, etc. for everything other than the final exam
  - If you use a portion of someone else's code (which is fine by me) make in-line acknowledgement in the comments of your code
  - Beware, that if your code starts to look like a collection of only other people's code, it's unlikely that the Final Exam will go well

Challenges

- Multiple student backgrounds and multiple topics mean that some students may feel like they would benefit from more challenging material… have no fear ;)
- I have collected some projects/questions from colleagues
- No guarantee that undertaking any advanced challenges will result in a better grade. Similarly, students may earn top marks in the course without ever looking at extra topics
- Potentially pick something on your own and discuss it with me, maybe even put together some lecture material and add it to the course

For the Proficient

- Some people will have excellent software/coding skills and will be able to quickly complete many of the in-class exercises. For those who consistently find themselves in this situation I offer an opportunity.
- The 2nd problem set — which is not yet assigned — will be very similar to exercises completed in class. I will offer extra credit for completing the problems using a different device. Playstation, mobile phone, dedicated GPU machine, etc.
- By first completing the in-class exercises ‘normal’ and then on a separate device, you’ll be prepared for when the 2nd problem set is assigned.

Expectations

- As graduate students, there is a rapidly growing importance for self-directed learning
- Software and hardware difficulties and solutions are the sole domain of each student. You can do all the projects on a PlayStation 4 w/ screenshots
- These are nominally advanced topics
  - I am excited to discuss new/other topics to cover in the course
  - We won't always have unassailable experts. So, discussion, and participation by individuals and groups is important
- Leave the course with at least 1 new tool that you can use in your research

Backup

Software Packages

- Some of the methods we will use in the course will require software packages that include:
  - Minimizers: for example BFGS, MIGRAD, SIMPLEX, etc.
  - Markov Chain Monte Carlo
  - Spline routines for interpolation, including basis splines (b-splines)
  - Multi-Variate Machine Learning: boosted decision trees, neural networks, support vector machines, etc. (we will for sure cover boosted decision trees)
- Other more specialized uses I will let you know about in advance of the lecture
  - Wavelets analysis needs deconvolution/decomposition methods and/or libraries
  - MultiNest nested sampling algorithm

More Specifically

- Below I will list the needed packages and some python options
- Plotting
  - I use ROOT from CERN, but that's because I used it extensively in my earlier research
  - Matplotlib is what the 'cool' kids use
- For Python users, I'm a big fan of "Jupyter" notebooks
  - Combination of both text fields, inline figures/plots display, and executable code
  - Great way to keep things organized
- Minimizer Routines
  - I normally use MINUIT2 (via iminuit)
  - SciPy has a minimize function with a bunch of algorithms and is more common nowadays

More Specifically

- Markov Chain Monte Carlo
  - I have used PyMC, but other packages such as MCMC, emcee, or Nestle look like better tools
- Multi-Variate Analysis (MVA)
  - I used the ROOT software from CERN
- Splines
  - SciPy has an interpolate function and other spline options
- Bayesian Inference Sampling - MultiNest
  - pymultinest
- Even if you're using python, you don't **need** any of the above mentioned *specific* packages, e.g. iminuit.

My Laptop

- As of Jan. 18, 2018 my laptop was setup as:
  - Mac OS Sierra (10.13.2)
  - Python 2.7.10
    - iPython 5.1.0
    - SciPy 0.13.01b
    - NumPy 1.8.0rc1
    - jupyter 4.2.1
  - ROOT 5.34/36
  - Homebrew (Mac package manager) 1.4.3
    - homebrew/core
    - homebrew/science
  - pip (python package manager) 9.0.1

My Laptop

- As of Feb. 5, 2017 my laptop was setup as:
  - Mac OS Sierra (10.12.3)
  - Python 2.7.10
    - iPython 5.1.0
    - SciPy 0.13.01b
    - NumPy 1.8.0rc1
    - jupyter 4.2.1
  - ROOT 5.34/36
  - Homebrew (Mac package manager) 1.1.8
    - homebrew/core
    - homebrew/python (not used though)
    - homebrew/science
  - pip (python package manager) 9.0.1